FOSDA: A Hybrid Disaggregated HPC Architecture based on Distributed Nanoseconds Optical Switches

Xiaotao Guo
Institute of Photonic Integration
Eindhoven University of Technology
Eindhoven, the Netherlands
e-mail: x.guo@tue.nl

Xuwei Xue
Institute of Photonic Integration
Eindhoven University of Technology
Eindhoven, the Netherlands
e-mail: x.xue.1@tue.nl

Bitao Pan
Institute of Photonic Integration
Eindhoven University of Technology
Eindhoven, the Netherlands
e-mail: b.pan@tue.nl

Fulong Yan
Institute of Photonic Integration
Eindhoven University of Technology
Eindhoven, the Netherlands
e-mail: f.yan@tue.nl

Georgios Exarchakos
Institute of Photonic Integration
Eindhoven University of Technology
Eindhoven, the Netherlands
e-mail: G.Exarchakos@tue.nl

Nicola Calabretta
Institute of Photonic Integration
Eindhoven University of Technology
Eindhoven, the Netherlands
e-mail: N.Calabretta@tue.nl

Abstract— **Aiming at solving the issues of low resource utilization and high operational cost in current node-centric High Performance Computing (HPC) architecture, we present and investigate a novel hybrid disaggregated HPC architecture based on nano-seconds fast optical switch (FOSDA). The fast optical switch connects Central Processing Unit (CPU) and memory nodes for the communication of high bandwidth and low latency, while storage nodes are interconnected by electrical packet switches. The performance of the FOSDA in terms of workload acceptance rate, resource utilization, power consumption and capital/operational cost is numerically assessed and compared with current node-centric HPC architectures. Compared with a node-centric HPC architecture of 320 nodes, FOSDA performs 36.6% higher CPU, and 21.5% higher memory resource utilization with 45.5% less active hardware resource, as well as 46.8% less power consumption. When scaling the HPC network to 2304 nodes, FOSDA also achieves 33.6% higher CPU and 48.5% higher memory utilization while saving 50.4% power consumption. In the cost analysis, FOSDA decreases operational cost by 46.7% with only 19.8% more capital cost.**

Keywords*- HPC network; disaggregated architecture; fast optical switch.*

I. INTRODUCTION

Facing the rapidly increasing diversity and scale of big data computing applications, HPC system has evolved from Symmetric Multi-Processing (SMP) architecture into cluster architecture consisting of thousands of computing nodes. With a tenfold performance increase every four years [1], the world's fastest HPC architecture Summit supports up to 4608 nodes [2]. All the hardware resources (i.e., CPU, memory, and network) are closely coupled in the computing nodes of the current HPC system (named node-centric architecture). Yet, various scientific computing applications have different resource requirements [3]. Memory intensive applications require sufficient memory resource to realize the in-memory and parallel data processing, while network intensive applications rely on network resources to implement dense inter-node communication. The tethered resources in computing nodes may lead to waste of specific resources when other resources are mostly used. These wasted resources also consume a large amount of power and operational cost. In addition, node-centric architecture also increases the upgrade cost. The whole node needs to be upgraded when only a specific hardware component comes to the end of the lifecycle. Therefore, it is necessary to develop a novel HPC architecture to provide more flexible resource provision, better performance for various applications, and more cost-efficient maintaining.

Recently, a promising disaggregated architecture is proposed for solving issues of low resource utilization, high power consumption, and high operation cost in current node-centric architectures [4]-[6]. In the disaggregated architecture, the on-board data bus connecting hardware in each computing node is replaced by the network interconnection. There have been several studies to implement the disaggregated architecture. The Rack Scale Design (RSD) from Intel sets up the independent storage resource management system [7], but CPU and memory are still fixed in the computing node. A disaggregated memory system is emulated based on the Xen hypervisor [8], while a remote memory paging system was designed based on the (Remote Direct Memory Access) RDMA protocol [9]. Meanwhile, a distributed operating system was also developed for the disaggregated architecture [10]. These solutions are based on current multi-tier electrical networks for thousands of nodes interconnection. However, the multistage switching in electrical switch operating at a high data rate results in high cost, O/E power consumption, and node-to-node latency. Some networks like Dragonfly [11] can reduce the latency and cost of the HPC network based on high radix switches, while having disadvantages of limited bandwidth and resiliency. Some efforts seek the optical network of high bandwidth and low latency for the disaggregated architecture. Compared with current electrical networks, the optical switching technologies are transparent to the data rate and packet protocol. Field-Programmable Gate Array (FPGA) based programmable Network Interface Card (NIC) and Optical Circuit Switch (OCS) of large radix and high data rate were developed in [12][13] for the hardware nodes interconnection. Based on hybrid OCS and electrical switches, a disaggregated architecture “dReDBox” was proposed in [14]. However, it has been demonstrated in [15] that the network latency has more impact to the performance of the disaggregated network than the bandwidth, and sub-microsecond network latency is necessary for the hardware nodes communication. The milliseconds reconfiguration of OCS may degrade the performance and flexible resource provision of the disaggregated architecture. To reduce the network switching time, an Optical Packet Switch (OPS) named Hipoλaos was proposed based on the tunable wavelength converter [16], but this structure could have some practical implementation issues such as high-speed operation and format-transparent operation. Therefore, a scalable interconnection network of high bandwidth and low latency for the disaggregated HPC system remains an open research problem to be solved to provide flexible resource provision and reduce operational cost.

In this work, we present a novel disaggregated architecture FOSDA based on distributed nanoseconds Fast Optical Switches (FOS) for the HPC system. Instead of multi-tier electrical networks in the current node-centric architecture, a novel flat optical interconnect network of high bandwidth is applied in the FOSDA. Exploiting the optical switch of data rate and packet protocol transparence, the FOSDA can provide a data rate of up to hundreds of gigabits for hardware communication. Meanwhile, considering the low latency requirement of the disaggregated architecture, the FOSDA is designed exploiting Semiconductor Optical Amplifier (SOA) based optical switching technologies. Benefiting from nanoseconds switching time, hardware nodes in the FOSDA are interconnected with an interconnect network of low latency and fast reconfiguration, and packet contentions are solved by the optical flow control protocol without optical buffer. Based on the distributed structure of the FOS and parallel processing for each channel, the FOSDA is scalable to interconnect thousands of hardware nodes. The FOSDA performance is assessed in terms of workload acceptance rate, resource utilization and power consumption. The evaluation results show that the FOSDA achieves a 13% higher acceptance rate, up to 36.6% higher CPU, and 21.5% higher memory resource utilization, as well as 46.8% less power consumption, compared with node-centric HPC architectures. With a large network scale of 2304 nodes, FOSDA performs 33.6% higher CPU utilization, 48.5% higher memory utilization, and 50.4% less power consumption. In addition, the capital and operational cost of FOSDA are also investigated and compared with node-centric HPC architectures. It is demonstrated that FOSDA requires 46.7% less operational cost with only 19.8% more capital cost.

The paper is organized as follows. In Section II, the FOSDA architecture and the system operation are described. Section III reports two node-centric architectures as comparison benchmarks, and traffic statistics of two benchmark networks are analyzed to generate workloads in the comparison. The performance of FOSDA are investigated and compared with two node-centric architectures in Section IV. Section V concludes the paper by summing up the most important results.

II. FOSDA HPC NETWORK AND OPERATION

The proposed FOSDA HPC network based on distributed nanoseconds FOS and optical flow control is depicted in Figure 1. It consists of *N* racks and each rack contains *N* CPU nodes (CN), *N* memory nodes (MN), and *M* storage nodes (SN). These hardware resources are disaggregated into specific resource pools. The FOS for intra-rack communication (RFOS) connects all CPU nodes and memory nodes in the same rack to achieve high bandwidth and low latency CPU-memory communication. About latency insensitive storage traffic, an electrical packet switch (EPS) is utilized to interconnect the memory nodes and storage nodes. CPU nodes *i* (CN_i) across different racks are interconnected by the $CFOS_i$ and, similarly, memory nodes *i* (MN_i) across different racks are interconnected by the $MFOS_i$ (i = 1, …, N). As illustrated in Figure 2, an address processor and a flow controller are integrated with the hardware node in the FOSDA network. The address processor receives the resource allocation table from resource manager of FOSDA, and forwards the traffic among

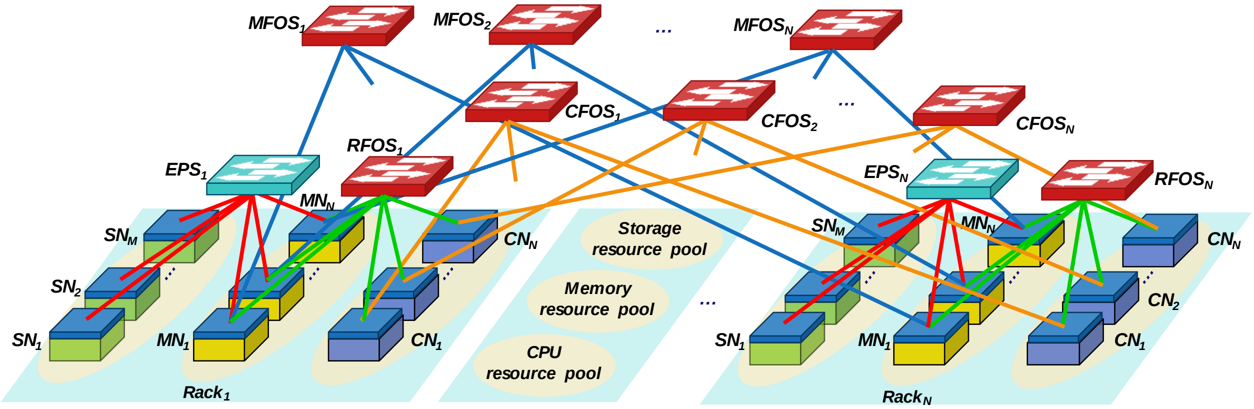


Figure 1. The FOSDA architecture.

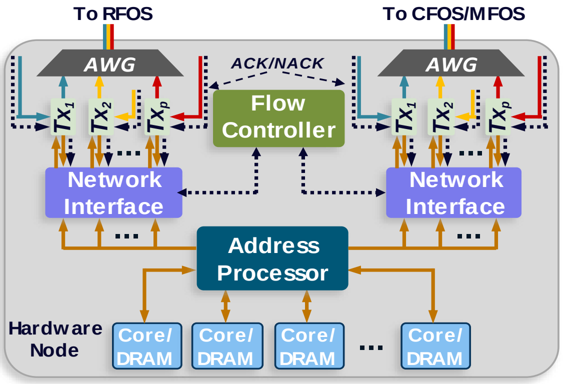


Figure 2. Fuctional blocks of the hardware node.

different resource nodes. Meanwhile, there is a flow controller in the hardware node to solve the packet contention of buffer-less optical switch [17]. The flow controller processes acknowledgement (ACK) or non-acknowledgement (NACK) signals from the FOS. The ACK signal represents the successful transmission, whereas the NACK signal means failure transmission due to packet contention. If receiving NACK signal, the flow controller sends the instruction to retransmit the packet. Both the intra-rack and inter-rack interfaces consist of *p* Wavelength-Division Multiplexing (WDM) transceivers. The *N* CPU (memory) nodes in each rack are divided into *p* groups, and each group has *F* nodes. Combining 1x*F* switch in the FOS, every TX in the hardware node serves for *F* potential destination hardware instead of *N*. Note that CPU node also contains a small memory to keep the operating system running. For the traffic between CPU nodes and memory nodes, according to the resource allocation table, the address processor maps the virtual memory address to the physical memory node address and inserts the destination address in the packet head of the CPU instruction. Then the optical packet is forwarded to RFOS/CFOS according to the address of the destination memory node. For the communication between the CPU node and the storage node, the instruction sent by the CPU node is processed by the memory controller in the memory node at first. Based on the instruction, the memory node starts reading/ writing data from/to the storage node via EPS. After finishing, the memory node sends the reply packet to the CPU node. For the traffic between memory nodes (e.g., live migration), CPU node distributes instructions to involved memory nodes first, and then memory nodes can keep on communicating without processing of CPU node.

The communication between CPU nodes and memory nodes in the same rack only crosses a single hop through the RFOS. It is also one hop for the communication between CPU/memory nodes at different racks if they are connected by the same CFOS/MFOS, while at most three hops are required to connect CPU nodes and memory nodes in the different racks. The number of hops is determined by the location of the requested data. The resource manager in the FOSDA system allocates the available memory nodes to cache the data. Based on the data request from the CPU nodes, the resource manager sets the path for the request to destination memory node. In each rack, the RFOS can support *N* CPU/memory nodes based on the *N* parallel processing modules. Meanwhile, the CFOS/MFOS interconnects the N CPU/memory nodes in N different racks. In total, *N* RFOSs are required for intra-rack communication, whereas *N* CFOS and *N* MFOS are required for inter-rack communication in the FOSDA system of *N* racks. Based on 3*N* FOS and *N* EPS, hardware resources of up to *N*×(2*N*+*M*) can be interconnected, and thus the scalability of FOSDA is guaranteed exploiting the FOS of moderate port counts.

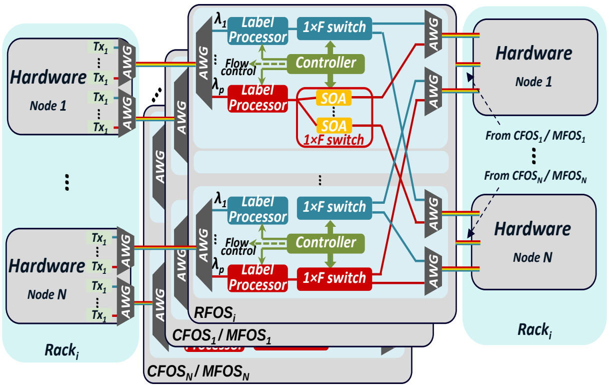


Figure 3. Schematic of the FOS.

The architecture of the FOS is illustrated in Figure 3. FOS processes the multiple WDM input packets from different resource nodes in parallel. The packets coming from the CPU/memory node i are processed by the label processor in the module *i*, where *i* = 1, ···, *N*. The optical label contains the packet destination and it is processed by the FPGA based optical switch controller. Meanwhile, the packet payload is broadcasted to the possible outputs via the 1x*F* SOA based nanoseconds optical gates. According to the lookup table distributed by the resource manager system and the optical label entry, the switch controller sets on/off the optical gates to forward or block the payload to the output destinations. If a contention happens, the packet with the highest priority is forwarded to the output, while the other packets are blocked. Then the switch controller sends the ACK/NACK signals back to the corresponding CN and MN nodes. The prototype of FOS was implemented in [18] including SOA based switch and FPGA based switch controller, which shows the feasibility of FOS. Due to the modularity and distributed processing, the FOS reconfiguration time is port count independent, allowing for low latency operation even for large scale interconnect networks.

III. Simulation Setup

We investigate the workload acceptance rate, resource utilization, required hardware amount, and power consumption of FOSDA under different node-centric architectures and network scales. The statistic number of workload requests per hour is based on the Poisson distribution. Workload acceptance rate represents the rate of accepted workload requests in all the workloads requests each minute, while the request rate is defined to represent the average number of workload requests per hour. Two node-

TABLE I. POWER AND COST OF COMPONENTS IN HPC ARCHITECTURE.

Components	**Specifications**		
	Type	***Power (W)***	***Cost ($)***
AMD Athlon MP2000+ processor	Idle Max	115 161	149
Intel Xeon E5-2660	Idle Max	116.4 194	1329
Memory	1G 32G 96G	0.373 11.85 35.55	6.5 209 637
NIC	Wulfkit3 10Gb/s 40Gb/s 56Gb/s	14 7 10.6 11.2	180 102 338 415
Transceiver	10Gb/s 40Gb/s 56Gb/s	1 3.5 4	18 59 84
Disk	HDD	6	154
Mellanox SX6536 Switch	648ports	9073	62,125
EPS	---	2/port	20/port
FOS	12ports 18ports 48ports	77 126 489	1140 2509 17612

centric HPC architectures of different network scales are considered as benchmarks in the comparison: HPC2N [19] and iDataPlex [20]. Workloads in the comparison is realistic traces from two benchmark HPC networks. Based on workload traces from two node-centric HPC architectures, the request rates are set to 17.44 for HPC2N and 26.46 for iDataPlex respectively. The simulation is based on CloudSimPy framework [21], and hardware is a workstation of 2 Intel Xeon Gold 5118 12 cores processors, 128GB memory, and NVIDIA Quadro 16GB P5000. The operation duration of 2880 minutes is set in the comparison. The fiber distance between hardware nodes and FOS is set to 20m. Based on the FOS reconfiguration time of 43.4ns, a Round-Trip Time (RTT) of FOSDA is 243.4ns that represents the minimum latency a packet may experience. The packet processing time in the hardware node is taken as 80ns [22], including the address processing and network protocol encapsulation. The bandwidth per transceiver is set to 40Gb/s in the FOSDA. The power consumption of each architecture is calculated by the sum of component power based on an additional model [23]. The power and cost of components in diverse HPC architectures are reported in Table I, according to academic studies [24]-[27] and current commercial products [28]-[30]. Based on analyses in [26] [27], the cost of 12-port FOS is $1140, while $17612 for 48-port FOS. Note that the cost of FOS can be further reduced by integrating the FOS processing modules into Application Specific Integrated Circuit (ASIC) chip.

The performance of the FOSDA is compared with two node-centric architectures in Section IV.A. The HPC2N consisting of 120 nodes (240 cores and 120GB memory in total). Based on a WulfKit 3 Scalable Coherent Interface (SCI) network, the computing nodes in the HPC2N are connected as a three-dimensional torus switching topology of 4×5×6 grid. The network consisting three SCI rings via Peripheral Component Interconnect (PCI) bus, providing 5.4Gb/s bandwidth per port. In this comparison, the FOSDA consists of 12 racks, supporting up to 144 nodes. To keep the same amount of hardware as the HPC2N, each rack has 10 CPU/memory nodes. The splitting ratio F equals 4, and transceiver amount for CFOS/MFOS is 3. The iDataPlex has a larger network scale than HPC2N, including 320 nodes (2560 cores and 10240 GB memory in total). The network interconnection of the iDataPlex cluster is based on the high-performance FDR InfiniBand network, in which a Mellanox SX6536 FDR 648-ports InfiniBand Director Switch is deployed. The FOSDA is set as 18 racks for the comparison with the iDataPlex. With the splitting ratio of 6 and transceiver amount of 3 in FOSDA, there are 2 racks consisting of 16 CPU/memory nodes to keep the hardware amount the same as iDataPlex. In Section IV.B, the scalability of FOSDA is investigated under the scale of 48 racks and 2304 nodes. The transceiver amount p and splitting ratio F of the optical switch equal 6 and 8 respectively. The node-centric architecture in this case is scaling iDataPlex cluster out to 2304 nodes. The network interconnection is based on Leaf Spine network topology, consisting of 48 96-ports switches and 8 128-ports switches. Finally, in Section IV.C, the capital and operational costs of FOSDA are analyzed and compared with two node-centric architectures under specifications shown in Table I.

The Cumulative Distribution Function (CDF) of workload statistic applied in assessments is shown in Figure 4. The workload traces are from two benchmark node-centric networks. For various hardware resources in HPC networks, storage and network resources are usually sufficient to serve workload requirement, and the bottleneck is often from the performance of CPU and memory. Therefore, the traffic statistics in assessments consist of the required amount of core and memory size as well as running time. The statistic

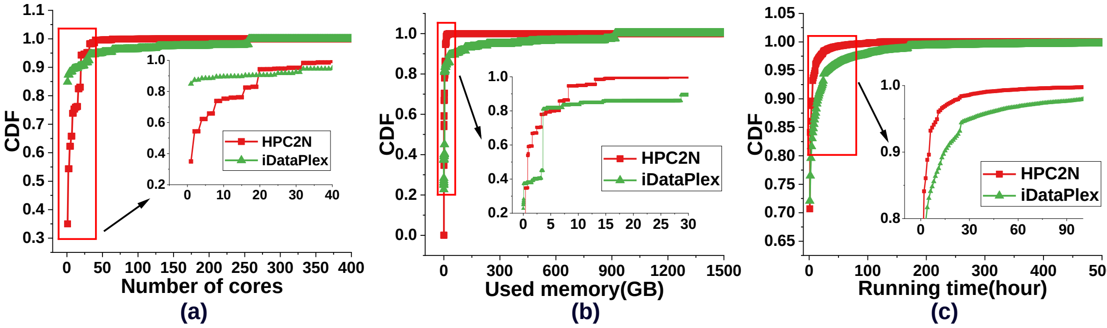


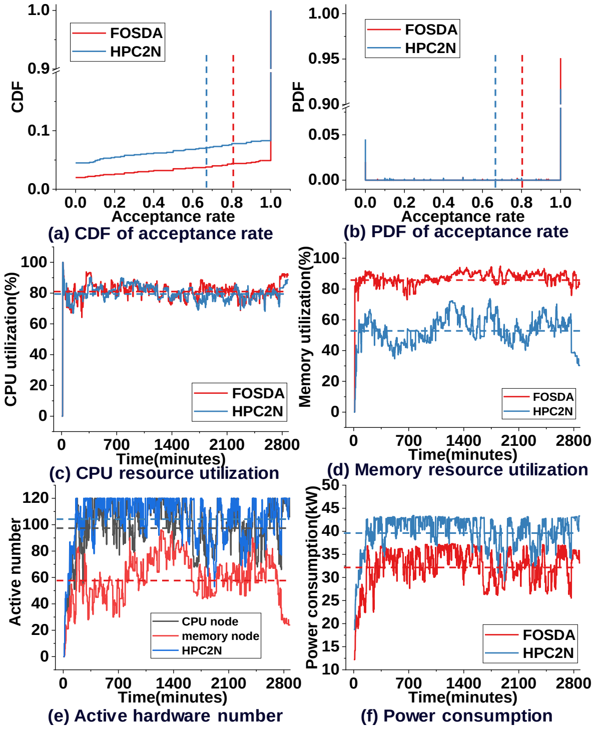
Figure 4. CDF of (a) CPU demand, (b) memory demand, and (c) running time.

of CPU demand in Figure 4(a) show that over 90% workloads have a CPU requirement of less than 50 cores in the both two architectures. Although more than 80% workloads require the CPU resource of less than 4 cores, iDataPlex also has more workloads requiring more than 250 cores CPU resource. Meanwhile, workloads have a more diverse demand of CPU resource in the HPC2N. Based on the CDF of memory demand illustrated in Figure 4(b), the memory demand in HPC2N mainly ranges from 0 and 17GB. Due to a larger network scale, iDataPlex also has 8.5% workloads with a memory demand of more than 100GB. It is shown in Figure 4(c) that more workloads require longer running time in the iDataPlex, while more than 60% workloads have a running time of less than 2 hours in two HPC networks.

IV. Assessment Results and Discussion

A. *Comparison with Two Node-centric Architectures*

Figures 5 and 6 show the workload acceptance rate, resource utilization, active hardware number, and power consumption of the FOSDA compared with two node-centric architectures under the realistic request rate. The comparison results between FOSDA and HPC2N are shown in Figure 5. As depicted in Figure 5(a) and (b), some workload requests are blocked in both the FOSDA and HPC2N due to the limited hardware resources. Despite this, the FOSDA can accept the workload requests with 80.7%, while the acceptance rate of HPC2N is 67.7%, which indicates that with the same amount of hardware, the FOSDA accepts 13% more workload requests. The reason is that, based on the independent resource allocation and fast network interconnection, the FOSDA can minimize the idle resource wasted, and has more available resources to deploy more workload request. Meanwhile, the average CPU resource utilization is 81.1% and 79.7% for FOSDA and HPC2N, respectively, as shown in Figure 5(c). The average memory resource utilization is 86.5% in FOSDA, which is also 33.4% higher compared with HPC2N in Figure 5(d). This is because the FOSDA provides more flexible resource to serve workloads with different requirements and maximize the utilization of each CPU/memory node. With much higher resource utilization and more powerful performance exploiting nanoseconds FOS, the FOSDA requires less hardware to deploy workload requests. Summarizing the active hardware resource in Figure 5(e), FOSDA only needs 97.6 CPU nodes and 58.4 memory nodes on average while HPC2N requires 103.9 computing nodes. Moreover, idle resource nodes in FOSDA are transferred into sleep mode that consume less power. As shown in Figure 5(f), the HPC2N consumes the power of 39.6kW and the FOSDA uses 18.7% less power than HPC2N.

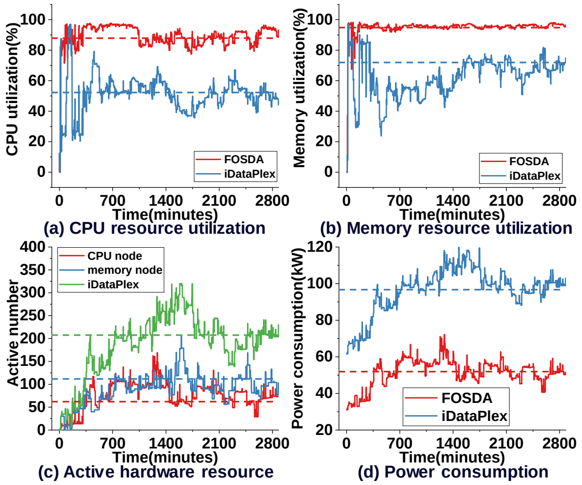


(a) CDF of acceptance rate (b) PDF of acceptance rate

(c) CPU resource utilization (d) Memory resource utilization

(e) Active hardware number (f) Power consumption

Figure 5. Comparison results between FOSDA and HPC2N.



(a) CPU resource utilization (b) Memory resource utilization

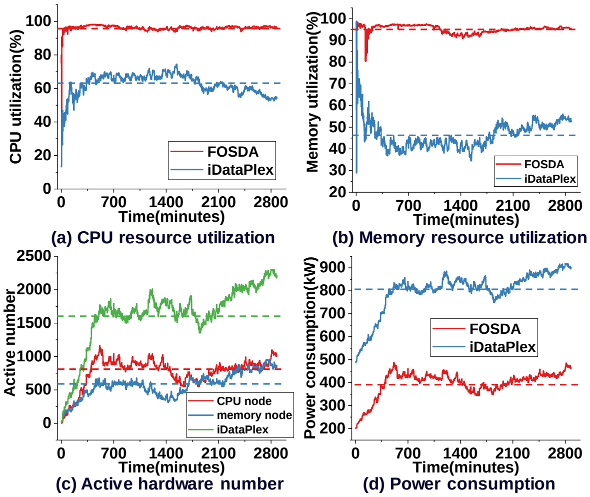
(c) Active hardware resource (d) Power consumption

Figure 6. Comparison results between FOSDA and iDataPlex.

The iDataPlex has a larger network scale and more hardware resources than the HPC2N. Thus, under the realistic request rate of 26.46, all the workload requests are accepted in FOSDA and iDataPlex. In the comparison of the CPU utilization, FOSDA achieves 88.7% utilization, as shown in Figure 6(a), which is 36.6% higher than the iDataPlex one. Meanwhile, Figure 6(b) shows that the FOSDA also obtain 21.5% higher memory resource utilization (FOSDA 93.4% while iDataPlex 71.9%). Benefiting from fully exploiting available resources in each resource node, workloads with diverse resource requirements achieve a better resource utilization in the FOSDA than the specific hardware (CPU or memory) intensive workloads. Figure 6(c) shows that FOSDA also needs less active hardware resources than iDataPlex. FOSDA uses 62.7 CPU nodes and 112.3 memory nodes while iDataPlex requires 206.2 computing nodes on average. In terms of average power consumption, FOSDA consumes 51.1kW while iDataPlex consumes 96.1kW, which achieves 46.8% power saving, as shown in Figure 6(d). Under workload requests with diverse requirements, FOSDA minimizes the waste of the active hardware, and achieves the largest power saving.

B. Scalability of FOSDA Architecture

To investigate the scalability of the FOSDA architecture, we consider a network scale of 2304 nodes in this section, and the request rate in this case is set to 200. With more available hardware, all the workload requests are accepted in both FOSDA and node-centric architectures. The comparison results of resource utilization, active resource number, and power consumption are reported in Figure 7. The resource utilization of CPU and memory are shown in Figure 7 (a) and (b). The CPU resource utilization for FOSDA is 96.1%, which is 33.6% higher than the iDataPlex. Meanwhile, the memory resource utilization for FOSDA is 95.1%, while for iDataPlex is 46.6%. Considering the comparison of the resource utilization in several cases, it is shown that the node-centric architecture cannot achieve high CPU and memory utilization simultaneously. This is because the CPU and memory are closely coupled in the node-centric architecture. When CPU (memory) achieves high utilization (less available), the available resource of memory (CPU) in the same computing node of node-centric architecture is wasted (low utilization). In the contrast, the FOSDA can achieve both high CPU and memory utilization based on the independent resource allocation. In response to the increased workload requests, there are more running hardware resources, as illustrated in Figure 7(c). The iDataPlex has 1618 computing nodes running in average, whereas the FOSDA only requires 769 CPU nodes and 574 memory nodes in total. Moreover, as presented in Figure 7(d), the power consumption of FOSDA is 397.6kW with 2304 nodes, which is 50.4% less compared with the iDataPlex.



(a) CPU resource utilization

(b) Memory resource utilization

(c) Active hardware number

(d) Power consumption

Figure 7. Comparison results under network scale of 2304 nodes.

It is demonstrated in numerical assessments that the FOSDA architecture outperforms current node-centric HPC networks regardless of workloads and network scales. This is because that the FOSDA architecture provides flexible resource provision and avoid the bottleneck of low speed peripheral bus in node-centric networks. Exploiting the flat and fast network interconnection based on FOS, the overhead of access memory node is minimized.

C. Capital and Operational Cost Comparison

Besides the operational performance, the capital and operational cost of FOSDA are also investigated and compared with the node-centric HPC architectures. Considering the small probability of hardware fault and diverse options of the hardware upgrade, we assume that all the hardware work in the normal state in the evaluation. In this section, the operation cost of FOSDA is compared with the node-centric architectures based on the realistic workload request rate. According to the industrial electricity price reported by the European Commission [31], the average power price is $0.11/kWh. Based on the subcomponents cost in Table I and power consumption results in numerical assessments, the capital and operational costs of FOSDA and node-centric architectures are shown in Table II, in which the operation cost is calculated for one year.

TABLE II. CAPITAL AND OPERATIONAL COST OF FOSDA AND NODE-CENTRIC ARCHITECTURES.

Architectures		**Cost**	
		Capital cost (k$)	***Operation Cost/year (k$)***
FOSDA	up to 144nodes up to 324nodes	346.8 1388.3	30.6 48.7
HPC2N	120 nodes	223.4	37.6
iDataPlex	320 nodes	1114	91.3

The maximum hosted node of the FOSDA is different to the node amount of node-centric architectures under a network scale. For the fairness of comparison, the FOSDA keeps the same hardware amount as node-centric architectures in the comparison. It is shown that 12 racks scale FOSDA (up to 144 nodes) saves 18.6% operational cost per year compared with HPC2N with 120 nodes, while requiring 35.6% higher capital cost (FOSDA 346.8k$ and HPC2N 223.4k$). Meanwhile, 18 racks scale FOSDA of (up to 324 nodes) requires 46.7% less operational cost than iDataPlex, at 19.8% higher capital cost (274.3k$). Those results indicate that, as the HPC network scales, the operational cost of FOSDA increases much slower than the one of the node-centric architectures. Balancing the capital and operation cost of the HPC system, it is demonstrated that FOSDA outperforms the current node-centric architectures, especially for large scale HPC system.

V. CONCLUSIOIN

In this work, we presented a novel disaggregated HPC architecture FOSDA based on distributed nanoseconds optical switches that connects the disaggregated resources by a flat scalable optical interconnect network of high bandwidth and low latency. Numerical assessments show that, compared with node-centric HPC architectures, FOSDA can accept up to 13% more workload requests, while achieving up to 36.6% higher CPU and 21.5% higher memory utilizations with 45.5% less active hardware resources. In addition, FOSDA saves 46.8% power consumption compared with node-centric HPC architecture of 320 computing nodes. With the increment of workload requests and network scale, FOSDA presents more advantages than node-centric architecture. FOSDA obtains 33.6% higher CPU and 48.5% higher memory utilization while saving 50.4% power consumption under a network scale of 2304 nodes. Moreover, compared with the node-centric HPC architectures, FOSDA requires 46.7% less operational cost with only 19.8% higher capital cost.

REFERENCES

[1] F. Karinou, I. Roudas, K. Vlachos, B. Hemenway, and R. Grzybowski, "Influence of transmission impairments on the OSMOSIS HPC optical interconnect architecture," J. Lightw. Technol., vol. 29, no. 21, pp. 3167–3177, 2011.

[2] Summit, https://www.olcf.ornl.gov/olcf-resources/compute-systems/summit, Retrieved: November, 2018

[3] R. Cheveresan, M. Ramsay, C. Feucht, and I. Sharapov, "Characteristics of workloads used in high performance and technical computing," Proceedings of the 21st annual international conference on Supercomputing, New York, USA, pp. 73–82, 2007.

[4] R. Lin, Y. Cheng, M. D. Andrade, L. Wosinska, and J. Chen, ''Disaggregated data centers: Challenges and trade-offs,'' IEEE Commun. Mag., vol. 58, no. 2, pp. 20–26, Feb. 2020.

[5] O. C. Project. The Open Compute server architecture specifications [Online], http://www.opencompute.org/. Retrieved: November, 2011

[6] P. Grun, "Introduction to infiniband for end users," [Online]. Retrieved: June, 2017, from http://www.mellanox.com/pdf/whitepapers/Intro_to_IB_for_End_Users.pdf. Accessed on: Jun. 10, 2017.

[7] Intel, "New photonic architecture promises to dramatically change next decade of disaggregated rack scale server designs," [Online], https://newsroom.intel.com/news-releases/intel-facebook-collaborate-on-future-data-center rack-technologies. Retrieved: December, 2016.

[8] K. Lim, et al. "System-level implications of disaggregated memory," in Proc. IEEE Int. Symp. High-Perform. Comput. Archit., Washington, USA, pp. 1–12, 2012.

[9] J. Gu, Y. Lee, Y. Zhang, M. Chowdhury, and K. Shin, "Efficient memory disaggregation with infiniswap," in Proc. 14th USENIX Symp. Netw. Syst. Design Implement. (NSDI), Carlsbad, USA, pp. 649–667, 2017.

[10] Y. Shan, Y. Huang, Y. Chen, and Y. Zhang, "Legoos: A disseminated, distributed OS for hardware resource disaggregation," in 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI 18). Carlsbad, CA, pp. 69–87, 2018.

[11] J. Kim, W. Dally, S. Scott, and D. Abts, "Technology-driven, highlyscalable dragonfly topology," SIGARCH Comput. Archit. News, vol. 36, pp. 77–88, 2008.

[12] Y. Yan, et al. "All-optical programmable disaggregated data center network realized by FPGA-based switch and interface card," J. Lightwave Technol., vol. 34, no. 8, pp. 1925–1932, 2016.

[13] G. Zervas, H. Yuan, A. Saljoghei, Q. Chen, and V. Mishra, "Optically disaggregated data centers with minimal remote memory latency: technologies, architectures, and resource allocation [Invited]," J. Opt. Commun. Netw., vol. 10, no. 2, pp. A270–A285, 2018.

[14] M. Bielski et al., "dReDBox: Materializing a full-stack rack-scale system prototype of a next-generation disaggregated datacenter," Design, Automation & Test in Europe Conference & Exhibition (DATE), Germany, pp. 1093-1098, 2018.

[15] P. Gao, et al. "Network requirements for resource disaggregation," in 12th USENIX Symp. on Operating Systems Design and Implementation (OSDI), Savannah, Georgia, pp. 249–264, 2016.

[16] N. Terzenidis, M. Moralis-Pegios, G. Mourgias-Alexandris, T. Alexoudi, K. Vyrsokinos and N. Pleros, "High-Port and Low-Latency Optical Switches for Disaggregated Data Centers: The Hipoλaos Switch Architecture," Journal of Optical Communications and Networking, vol. 10, no. 7, pp. 102-116, 2018.

[17] W. Miao and N. Calabretta, "Low latency and efficient optical flow control for intra data center networks," Opt. Express vol. 22, no. 1, pp. 427–434, 2014.

[18] X. Xue et al., "ROTOS: A Reconfigurable and Cost-Effective Architecture for High-Performance Optical Data Center Networks," Journal of Lightwave Technology, vol. 38, no. 13, pp. 3485-3494, 2020.

[19] HPC2N, https://www.hpc2n.umu.se/resources/hardware/seth. Retrieved: June, 2017.

[20] iDataPlex, https://www.pik-potsdam.de/services/it/hpc. Retrieved: January, 2019.

[21] CloudSimPy, https://github.com/RobertLexis/CloudSimPy. Retrieved: December, 2020.

[22] F. Yan, W. Miao, O. Raz and N. Calabretta, "OPSquare: A flat DCN architecture based on flow-controlled optical packet switches," Journal of Optical Communications and Networking, vol. 9, no. 4, pp. 291-303, 2017.

[23] M. Dayarathna, Y. Wen, and R. Fan, "Data center energy consumption modeling: a survey." IEEE Commun Surv Tutorials, vol. 18, no. 1, pp. 732–94, 2016.

[24] B. Giridhar, et al., "Exploring DRAM organizations for energy-efficient and resilient exascale memories," in Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal. SC, pp. 23:1–23:12, 2013.

[25] J. Aroca, A. Chatzipapas, A. Anta, and V. Mancuso, "A Measurement-based Analysis of the Energy Consumption of Data Center Servers." Proceedings of the 5th international conference on Future energy systems, UK, pp. 63-74, 2014.

[26] F. Yan, W. Miao, H. Dorren, and N. Calabretta, "On the cost, latency, and bandwidth of LIGHTNESS data center network architecture." International Conference on Photonics in Switching, Italy, pp. 130-132, 2015.

[27] X. Guo et al., "RDON: a rack-scale disaggregated data center network based on a distributed fast optical switch," Journal of Optical Communications and Networking, vol. 12, no. 8, pp. 251-263, 2020.

[28] AMD Athlon Dual-Core Processor [Online], https://www.amd.com/system/files/TechDocs/33425.pdf. Retrieved: January, 2007.

[29] Intel Xeon E5 Processor [Online], https://ark.intel.com/content/www/us/en/ark.html#@Processors. Retrieved: March, 2012.

[30] Mellanox SX6536 Switch, [Online], http://www.mellanox.com/related-docs/user_manuals/SX6536_user_manual.pdf. Retrieved: January, 2017.

[31] Energy prices and costs in Europe [Online], https://ec.europa.eu/energy/sites/ener/files/report_on_energy_prices_and_costs_in_europe_com_2020_951.pdf. Retrieved: October, 2020.